STAT2_HW2

Xingyu Chen

March 27, 2022

Total Score: 12/14

1 Homework #3

See Canvas for HW #3 assignment due date.

1.0.1 Problem B.1: Model Selection Criterion

In this lesson, we will perform both the full and partial F-tests in R.

We will use the Amazon book data.

https://raw.githubusercontent.com/bzaharatos/-Statistical-Modeling-for-Data-Science-Applications/master/Modern Regression Analysis / Datasets/amazon.txt

The data consists of data on $n = 325$ books and includes measurements of:
- aprice : The price listed on Amazon (dollars) - Lprice : The book's list price (dollars) - weight: The book's weight (ounces) - pages : The number of pages in the book - height : The book's height (inches) - width : The book's width (inches) - thick: The thickness of the book (inches) - cover: Whether the book is a hard cover of paperback. - And other variables...

I will include some data cleaning to get you started, although you don't have to use this exact code. We do want to remove NA and average out what we can beforehand. For all tests in this lesson, let $\alpha = 0.05$.

```
## Here is the data cleaning I mentioned. Again, feel free to explore this via your ow
url = paste0("https://raw.githubusercontent.com/bzaharatos/",
                    "-Statistical-Modeling-for-Data-Science-Applications/",
                    "master/Modern%20Regression%20Analysis%20/Datasets/amazon.txt")
amazon <- read.delim2(url)
df = data.frame(aprice = amazon$Amazon.Price, lprice = as.numeric(amazon$List.Price),
                pages = amazon$NumPages, width = amazon$Width, weight = amazon$Weight..o
                height = amazon$Height, thick = amazon$Thick, cover = amazon$Hard..Paper
df$lprice[which(is.na(df$lprice))] = mean(df$lprice, na.rm = TRUE)
df$weight[which(is.na(df$weight))] = mean(df$weight, na.rm = TRUE)
```

```r
df$pages[which(is.na(df$pages))] = mean(df$pages, na.rm = TRUE)
df$height[which(is.na(df$height))] = mean(df$height, na.rm = TRUE)
df$width[which(is.na(df$width))] = mean(df$width, na.rm = TRUE)
df$thick[which(is.na(df$thick))] = mean(df$thick, na.rm = TRUE)

head(df)
```

aprice	lprice	pages	width	weight	height	thick	cover
5.18	12.95	304	5.5	11.2	7.8	0.8	P
10.2	15.00	273	5.5	7.2	8.4	0.7	P
1.5	1.50	96	5.2	4	8.3	0.3	P
10.87	15.99	672	6	28.8	8.8	1.6	P
16.77	30.50	720	5.2	22.4	8	1.4	P
16.44	28.95	460	6.3	32	8.9	1.7	H

1.0.1.0.1 B.1. (a) The Model We want to determine which predictors impact the Amazon list price. Begin by fitting the full model.

Fit a model named lmod.full to the data with aprice as the response and all other rows as predictors. Then calculate the AIC, BIC and adjusted R^2 for this model. Store these values in AIC. full, BIC. fulL and adj.R2. full respectively. Q1: 2/2
Answer:

```r
df_num <- as.data.frame(sapply(df, as.numeric))
```

```
## Warning in lapply(X = X, FUN = FUN, ...): NAs introduced by coercion
```

```r
df_num$cover <- df$cover
head(df_num)
```

aprice	lprice	pages	width	weight	height	thick	cover
5.18	12.95	304	5.5	11.2	7.8	0.8	P
10.20	15.00	273	5.5	7.2	8.4	0.7	P
1.50	1.50	96	5.2	4.0	8.3	0.3	P
10.87	15.99	672	6.0	28.8	8.8	1.6	P
16.77	30.50	720	5.2	22.4	8.0	1.4	P
16.44	28.95	460	6.3	32.0	8.9	1.7	H

```r
df <- df_num
df <- na.omit(df)
```

```r
lmod.full <- lm(aprice ~ . , data = df)
AIC.full <- AIC(lmod.full)
BIC.full <- BIC(lmod.full)
```

```r
adj.R2.full <- summary(lmod.full)$adj.r.squared
AIC.full
```

```
## [1] 2105.488
```

```r
BIC.full
```

```
## [1] 2139.232
```

```r
adj.R2.full
```

```
## [1] 0.7245236
```

1.0.1.0.2 B.1. (b) A Partial Model

Fit a partial model to the data, with aprice as the response and Lprice and pages as predictors. Calculate the AIC, BIC and adjusted R^2 for this partial model. Store their values in AIC. part, BIC. part and adj.R2. part respectively.
Q2: 2/2
Answer:

```r
lmod.partial <- lm(aprice ~ lprice + pages , data = df)
AIC.partial <- AIC(lmod.partial)
BIC.partial <- BIC(lmod.partial)
adj.R2.partial <- summary(lmod.partial)$adj.r.squared
AIC.partial
```

```
## [1] 2105.26
```

```r
BIC.partial
```

```
## [1] 2120.257
```

```r
adj.R2.partial
```

```
## [1] 0.7203847
```

1.0.1.0.3 B.1. (c) Model Selection

Which model is better, Lmod. full or Lmod. part according to

$$\text{AIC}, \text{BIC}, and R_a^2$$

? Note that the answer may or may not be different across the different criteria. Save your selections as selected.model. AIC, selected.model. BIC, and selected.model.adj.R2 .
Q3: 1/2
Answer:

```r
selected.model.AIC <- AIC.partial
selected.model.BIC <- BIC.partial
selected.model.adj.R2 <- adj.R2.full
```

1.0.1.0.4 B.1. (d) Model Validation

Recall that a simpler model may perform statistically worse than a larger model. Test whether there is a statistically significant difference between lmod.part and Lmod. full. Based on the result of this test, what model should you use?

Q4: 1/2

Answer:

```
anova(lmod.partial, lmod.full)
```

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
311	14627.92	NA	NA	NA	NA
306	14179.70	5	448.2205	1.934533	0.0883967

I would choose the partial model because p-value greater than 0.05, is not statistically significant and indicates strong evidence for the null hypothesis.

1.0.2 Problem B.2

divorce is a data frame with 77 observations on the following 7 variables.
1. year : the year from 1920-1996 2. divorce : divorce per 1000 women aged 15 or more 3. Unemployed unemployment rate 4. femlab : percent female participation in labor force aged 16+ 5. marriage : marriages per 1000 unmarried women aged 16+ 6. birth : births per 1000 women aged 15-44 7. military : military personnel per 1000 population Here's the data: (I'll also include all data links in Canvas)

```
url = paste0("https://raw.githubusercontent.com/bzaharatos/",
             "-Statistical-Modeling-for-Data-Science-Applications/",
             "master/Modern%20Regression%20Analysis%20/Datasets/divusa.txt")
df_b <- read.delim2(url)
```

1.0.2.0.1 B.2 (a)

Using the divorce data, with divorce as the response and all other variables as predictors, select the "best" regression model, where "best" is defined using AIC. Save your final model as

$$Lm_{divorce}$$

.

Q5: 2/2

Answer:

```
df_num <- as.data.frame(sapply(df_b, as.numeric))
head(df_num)
```

year	divorce	unemployed	femlab	marriage	birth	military
1920	8.0	5.2	22.70	92.0	117.9	3.2247
1921	7.2	11.7	22.79	83.0	119.8	3.5614

year	divorce	unemployed	femlab	marriage	birth	military
1922	6.6	6.7	22.88	79.7	111.2	2.4553
1923	7.1	2.4	22.97	85.2	110.5	2.2065
1924	7.2	5.0	23.06	80.3	110.9	2.2889
1925	7.2	3.2	23.15	79.2	106.6	2.1735

```
df_b <- df_num
df_b<- na.omit(df_b)
```

```
library(AICcmodavg)
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 4.1.2
```

```
lm_full <- lm(divorce ~ . , data = df_b)
stepAIC(lm_full, direction = "both")
```

```
## Start:  AIC=70.41
## divorce ~ year + unemployed + femlab + marriage + birth + military
##
##              Df Sum of Sq    RSS     AIC
## - unemployed  1     1.925 162.12  69.330
## <none>                    160.20  70.410
## - military    1    22.231 182.43  78.417
## - year        1    33.199 193.40  82.912
## - marriage    1    90.468 250.66 102.884
## - femlab      1   113.214 273.41 109.572
## - birth       1   144.897 305.10 118.015
##
## Step:  AIC=69.33
## divorce ~ year + femlab + marriage + birth + military
##
##              Df Sum of Sq    RSS     AIC
## <none>                    162.12  69.330
## + unemployed  1     1.925 160.20  70.410
## - military    1    20.957 183.08  76.691
## - year        1    42.054 204.18  85.089
## - marriage    1   126.643 288.77 111.779
## - femlab      1   158.003 320.13 119.718
## - birth       1   172.826 334.95 123.203
```

```
##
## Call:
## lm(formula = divorce ~ year + femlab + marriage + birth + military,
##     data = df_b)
```

```
##
## Coefficients:
## (Intercept)         year       femlab     marriage        birth     military
##    405.6167      -0.2179       0.8548       0.1593      -0.1101      -0.0412
```

```r
lm_divorce <- lm(divorce ~ year + femlab + marriage + birth + military,
    data = df_b)
```

1.0.2.0.2 B.2 (b)

Using your model from part (a), compute the variance inflation factors VIFs for each $\widehat{\beta}_j, j = 1, \ldots, p$. Store them in the variable v. Also, compute the condition number for the design matrix, stored in k. Is there evidence that collinearity causes some predictors not to be significant? Explain your answer.

Q6: 2/2

Answer:

```r
library(car)
```

```
## Warning: package 'carData' was built under R version 4.1.2
```

```r
#summary(lm_divorce)
v <- vif(lm_divorce)
v
```

```
##      year    femlab  marriage     birth  military
## 42.948267 48.650935  2.624531  2.031677  1.358002
```

```r
k <- kappa(df_b[, c('divorce', 'year', 'femlab' ,'marriage' , 'birth' , 'military')])
k
```

```
## [1] 1344.989
```

Yes, year and femlab VIF is exceeding 5 or 10 indicates high multicollinearity between this independent variable and the others.

1.0.2.0.3 B.2 (c)

Remove the predictor with the highest VIF. Does that reduce the multicollinearity?

Q7: 2/2

Answer:

```r
lm_divorce_2 <- lm(divorce ~ year + marriage + birth + military,
    data = df_b)
v <- vif(lm_divorce_2)
v
```

```
##     year marriage    birth military
## 1.833706 2.331921 1.982541 1.125807
```

```r
k <- kappa(df_b[, c('divorce', 'year' ,'marriage' , 'birth' , 'military')])
k
```

```
## [1] 460.6529
```

Yes, it reduce the multicollinearity